Kryptografia z kluczem publicznym RSA.

Marek Zawadowski
Wydział Matematyki, Informatyki i Mechaniki
Uniwersytet Warszawski

Kwiecień 2005

Kryptografia z kluczem publicznym RSA opiera się na dwóch podstawowych założeniach:

1. Łatwo znaleźć duże liczby pierwsze (np. 100 cyfrowe);
2. Rozłożenie dużych liczb złożonych (np. 200 cyfrowych) na czynniki pierwsze przekracza obecne możliwości komputerów;

z których pierwsze jest *probabilistycznie* (od lata 2002 już niekoniecznie) a drugie *empirycznie* prawdziwe.

1 Kryptografia RSA

1. ***Opis ogólny:***
   (a) A i B to dwie strony, które chcą do siebie przesyłać wiadomości;
   (b) Każda strona ma klucz publiczny P_A, P_B znany wszystkim użytkownikom i klucz sekretny S_A, S_B znany tylko im;
   (c) $\mathcal{D}$ jest to zbiór dozwolonych wiadomości (np. ciągów o ustalonej długości złożonych z 0 i 1).
   (d) klucze definiują funkcje: $P_A, P_B, S_A, S_B : \mathcal{D} \to \mathcal{D}$ takie, że P_A jest odwrotna do S_A i P_B jest odwrotna do S_B, tzn $M = P_A S_A(M)$, $M = S_A P_A(M)$ dla $M \in \mathcal{D}$;
2. ***Tworzenie kluczy:***
   (a) Wybieramy dwie duże liczby pierwsze p i q.

(b) Obliczamy $n = p \cdot q$ i $\phi(n) = (p-1) \cdot (q-1)$.

(c) Wybieramy niedużą liczbę nieparzystą e względnie pierwszą z $\phi(n)$.

(d) Obliczamy liczbę $d < n$ taką, że

$$e \cdot d \equiv_{\phi(n)} 1$$

(takie d istnieje i jest jedyne).

(e) Publikujemy parę $P = (e, n)$ jako klucz publiczny.

(f) Trzymamy w sekrecie parę $S = (d, n)$ jako klucz sekretny.

3. ***Kodowanie i dekodowanie:***

(a) Naszą dziedziną jest $\mathcal{D} = Z_n = \{0, \ldots, n-1\}$.

(b) Kodowanie definiujemy tak: $P(M) = M^e \bmod n$, dla $M \in \mathcal{D}$.

(c) Dekodowanie definiujemy tak: $P(M) = M^d \bmod n$, dla $M \in \mathcal{D}$.

Uwagi. Liczbę sekretną d można łatwo obliczyć mając e i $\phi(n)$. Zatem, jeżeli uda nam się rozłożyć n na czynniki p i q to możemy obliczyć też $\phi(n) = (p-1) \cdot (q-1)$. Dlatego ważne jest dla powodzenie tego protokołu kryptograficznego, by rozłożenie n na czynniki pierwsze było (obliczeniowo) niemożliwe.

2 Elementy teorii liczb.

NWD i rozszerzony algorytm Euklidesa.

Algorytm ten działa jak algorytm Euklidesa znajdowania NWD dwóch liczb ale przy okazji liczy dodatkowe wartości, które będą nam przydatne później.

Lemat 2.1 *Niech $a, b \in N$ i co najmniej z liczb różna od 0. Wtedy $NWD(a, b)$ jest najmniejszą dodatnią liczbą w zbiorze całkowitych kombinacji liniowych a i b*

$$\{x \cdot a + y \cdot b : x, y \in Z\}.$$

Dowód: Zauważmy, że jeśli liczba c dzieli a i b więc dzieli też każdą kombinację liniową a i b.

Niech $d = NWD(a, b)$ a s będzie tą najmniejszą liczbą dodatnią ze zbioru $\{x \cdot a + y \cdot b : x, y \in Z\}$, $s = x \cdot a + y \cdot b$ oraz $q = \lfloor \frac{a}{s} \rfloor$. Wtedy

$$a \bmod s = a - q \cdot s = a - q \cdot (x \cdot a + y \cdot b) = (1 - q \cdot x) \cdot a + (-qy) \cdot b$$

Zatem a mod s kombinacją liniową a i b. Ponieważ a mod $s < s$ i s jest najmniejszą dodatnią kombinacją liniową a i b to a mod $s = 0$. Czyli $s|a$. Podobnie można pokazać, że $s|b$. A stąd $s|d$. Z drugiej strony, ponieważ s jest kombinacją liniową a i b i $d = NWD(a, b)$ to $d|s$. Zatem $s = d$.

Q.E.D.

Wniosek 2.2 *1. Niech $a, b, p \in N$ i $NWD(a, p) = NWD(b, p) = 1$. Wtedy $NWD(a \cdot b, p) = 1$.*

2. Niech $a, b, n \in N$, $NWD(a, b) = 1$, $a|n$ i $b|n$. Wtedy $a \cdot b|n$.

Dowód: Ad 1. Z Lematu 2.1 mamy $x_1, x_2, y_1, y_2 \in Z$ takie, że

$$x_1 a + y_1 p = 1 \quad x_2 b + y_2 p = 1$$

a mnożąc stronami te równości otrzymujemy

$$(x_1 x_1)ab + (y_1 b x_2 + y_2 a x + p y_1 y_2)p = 1$$

Zatem z Lematu 2.1 mamy $NWD(a \cdot b, p) = 1$.

Ad 2. Niech $a, b, n \in N$, $NWD(a, b) = 1$, $a|n$ i $b|n$. Wtedy istnieją n_1, n_2 takie, że

$$n = a n_1 = b n_2 \tag{1}$$

i Lematu 2.1 istnieją $x, y \in Z$ takie, że

$$xa + yb = 1.$$

Podstawiając $a = \frac{1-yb}{x}$ do (1) otrzymujemy

$$b n_2 = \frac{1 - yb}{x} n_1$$

$$b n_2 x = n_1 - y b n_1$$

$$b(n_2 x + n_1 y) = n_1 \tag{2}$$

Z (1) i (2) mamy

$$ab(n_2 x + n_1 y) = a n_1 = n.$$

A to oznacza, że $ab|n$.

Q.E.D.

Poniższy problem dotyczy znajdowania konkretnej kombinacji liniowej dwóch liczb równej NWD tych liczb.

Problem.

- Dane wejściowe: liczby naturalne a, b takie, że $a \geq b$;
- Wynik: liczby naturalne $d = NWD(a, b)$, x, i y takie, że $d = x \cdot a + y \cdot b$;

```
procedure euklides (a,b:integer; var d,x,y:integer);
begin
 if b=0 then begin d:=a; x:=1; y:=0 end
  else begin
        euklides(b, a mod b,d,x1,y1);
        x:=y1; y:=x1-(a div b)*y1;
       end;
end;
```

Ponieważ $a = (a \text{ div } b)b + (a \bmod b)$ jeśli

$$d = x_1 b + y_1 (a \bmod b)$$

to również

$$d = y_1 a + (x_1 - (a \bmod b) y_1) b.$$

A zatem powyższy algorytm jest poprawny.

Grupy skończone i arytmetyka modularna.

Grupą nazywamy trójkę $(G, *, e)$ gdzie $* : G \times G \to G$ jest działaniem dwuargumentowym, $e \in G$ oraz spełnione są warunki:

1. $a * (b * c) = (a * b) * c$ dla dowolnych $a, b, c \in G$ (*łączność*);
2. $e * a = a * e = a$ dla dowolnego $a \in G$ (e jest elementem *neutralnym*);
3. dla dowolnego $a \in G$ istnieje jedyny element $b \in G$ taki, że $a * b = e$; takie b oznaczamy a^{-1}; (istnieją elementy *odwrotne*);.

Jeśli $ab = ba$ dla dowolnych $a, b \in G$ to grupę nazywamy *abelową*. *Rzędem* grupy nazywamy liczbę elementów G i oznaczamy $|G|$. Jeśli rząd grupy jest skończony to grupę nazywamy *skończoną*. Często utożsamiamy zbiór G z grupą $(G, *, e)$, jeśli działanie grupowe $*$ i jedność e wynikają z kontekstu.

Przykłady grup.

1. $(Z, +, 0)$ - grupa addytywna liczb całkowitych.
2. $(Q, +, 0)$ - grupa addytywna liczb wymiernych.
3. $(Q^*, \cdot, 1)$ - grupa multyplikatywna dodatnich liczb wymiernych.

   Operacja mod n jest przemienna z dodawaniem i mnożeniem, dla $a, b \in Z$

   $$(a +_n b) = (a + b) \bmod n = ((a \bmod n) + (b \bmod n)) \bmod n =$$

   $$= (a + (b \bmod n)) \bmod n = (a \bmod n) + b) \bmod n$$

   oraz

   $$(a \cdot_n b) = (a \cdot b) \bmod n = ((a \bmod n) \cdot (b \bmod n) \bmod n =$$

   $$= ((a \cdot (b \bmod n) \bmod n = (a \bmod n) \cdot b) \bmod n$$

   Stąd otrzymujemy dwa kolejne przykłady grup skończonych.
4. $(Z_n, +_n, 0)$ - grupa addytywna modulo n ($Z_n = \{0, 1, \ldots, n-1\}$).
5. $(Z_n^*, \cdot_n, 1)$ - grupa multiplikatywna modulo n ($Z_n^* = \{a \in Z_n | NWD(n, a) = 1\}$). Z Wniosku 2.2 $\cdot_n$ jest dobrze określonym działaniem. Z Lematu 2.1, dla $a \in Z_n^*$ istnieją $x, y \in Z$ takie, że $xa + yn = 1$. Zatem $xa \equiv_n 1$ i $x = a^{-1}$.
6. Grupa permutacji zbioru n-elementowego S_n.
7. Grupa permutacji parzystych zbioru n-elementowego A_n.
8. Grupa addytywna $(V, 0, +)$, przestrzeni wektorowej $(V, 0, +, \cdot)$.

Grupa Z_n^* jest dla nas najważniejsza. Niech $\phi(n)$ rzędem grupy Z_n^*. Oczywiście, jeśli n jest liczbą pierwszą to

$$\phi(n) = n - 1,$$

jeśli $n = p \cdot q$ gdzie p i q są liczbami pierwszymi to

$$\phi(n) = (p-1) \cdot (q-1).$$

Można pokazać ogólniej, że dla dowolnego n

$$\phi(n) = n \prod_{p|n} (1 - \frac{1}{p})$$

gdzie p przebiega liczby pierwsze. Na przykład

$$\phi(63) = 63 \cdot (1 - \frac{1}{3}) \cdot (1 - \frac{1}{7}) = 63 \cdot (\frac{2}{3}) \cdot (\frac{6}{7}) = 35$$

Grupa G' nazywamy *podgrupą* grupy G jeśli $G' \subseteq G$ oraz G' jest jest zamknięte w G na działanie grupowe, jedność i elementy odwrotne.

Na przykład $(Z, +, 0)$ jest podgrupą $(Q, +, 0)$.

Twierdzenie 2.3 (Twierdzenie Lagrange'a) *Jeśli G' jest podgrupą grupy skończonej G to rząd G' dzieli rząd G.*

Niech a będzie elementem grupy G. Oznaczmy

$$a^{(k)} = \underbrace{a * \ldots * a}_{k \text{ razy}}$$

Wtedy zbiór

$$\{a^{(1)}, a^{(2)}, \ldots, a^{(n)}, \ldots\}$$

jest skończoną podgrupą G, oznaczaną przez $< a >$. Rząd $< a >$ nazywamy rzędem elementu a. Innymi słowy rząd a jest to najmniejsza taka liczba naturalna dodatnia k, że $a^{(k)} = e$.

Z twierdzenia Lagrange'a wynika, że jeśli G jest grupą skończoną to $a^{|G|} = e$, dla dowolnego $a \in G$. Stąd w szczególnym przypadku, dla grupy Z_n^* mamy

Twierdzenie 2.4 (Twierdzenie Euler'a) *Niech $a \in Z_n^*$. Wtedy*

$$a^{\phi(n)} \equiv_n 1.$$

A stąd

Twierdzenie 2.5 (Małe Twierdzenie Fermata) *Niech p będzie liczbą pierwszą oraz $1 \leq a < p$. Wtedy*

$$a^{p-1} \equiv_n 1.$$

Do znajdowania dużych liczb pierwszych potrzebne nam będzie jeszcze poniższe twierdzenie mówiące o pierwiastkach z 1 modulo n. Dowód tego twierdzenia jest nieco trudniejszy i zostanie on pominięty.

Twierdzenie 2.6 *Niech p będzie liczbą pierwszą i $e \geq 1$. Wtedy równanie*

$$x^2 \equiv_{p^e} 1$$

ma dokładnie dwa rozwiązania (pierwiastki z 1), $x = -1 = (p^e - 1)$ i $x = 1$.

Poniższe twierdzenie jest potrzebne do stwierdzenia poprawności protokołu RSA. Jest to jedna z wersji tzw. Twierdzenia chińskiego o resztach. Jeśli liczba $n = n_1 \cdot \ldots \cdot n_k$, oraz $n_1, \ldots, n_k$ są względnie pierwsze to grupa Z_n jest izomorficzna (='taka sama') z produktem kartezjańskim grup $Z_{n_1} \times \ldots \times Z_{n_k}$.

Twierdzenie 2.7 (Twierdzenie chińskie o resztach) *Niech $n_1, \ldots, n_k$ będą liczbami dodatnimi parami względnie pierwszymi, $n = n_1 \cdot \ldots \cdot n_k$, oraz $a_i \in Z_{n_i}$ dla $i = 1, \ldots, k$. Wtedy istnieje jedyna liczba $a \in Z_n$ taka, że*

$$a \equiv_{n_i} a_i$$

dla $i = 1, \ldots, k$

Dowód: Niech $m_i = n/n_i$. Z Wniosku 2.2 wynika, że $NWD(m_i, n_i) = 1$. Zatem istnieje element $(m_i^{-1} \bmod n_i)$, który jest odwrotnością $(m_i \bmod n_i)$ w grupie $Z^*_{n_i}$. Niech

$$c_i = m_i \cdot (m_i^{-1} \bmod n_i)$$

Kładziemy

$$a = (a_1 \cdot c_1 + \ldots + a_k \cdot c_k) \bmod n.$$

Zauważmy, że mamy

$$m_j \equiv_{n_i} 0$$

dla $i \neq j$. A stąd

$$a \equiv_{n_i} a_i \cdot c_i \equiv_{n_i} a_i \cdot m_i \cdot (m_i^{-1} \bmod n_i) \equiv_{n_i} a_i$$

Zatem wykazaliśmy istnienie a.

Pokażemy, że takie $a \in Z_n$ jest jedyne. Niech $b \in Z_n$ takie, że $b \equiv_{n_i} a_i$, dla $i = 1, \ldots, k$. Pokażemy, że $a = b$. Niech $c = b -_n a$. Wtedy $0 \leq c < n$. Ponadto $c \equiv_{n_i} 0$, zatem $n_i | c$, dla $i = 1, \ldots, k$. Ponieważ n_i są względnie pierwsze to z Wniosku 2.2 mamy, że $n = n_1 \cdot \ldots \cdot n_k | c$. Zatem o ile $c > 0$ to $c \geq n$. Ale $c < n$, więc $c = 0$, i $a = b$.

Q.E.D.

Przykład. Dane są liczby względnie pierwsze $n_1 = 3$, $n_2 = 4$, $n_3 = 5$ i trzy inne liczby mniejsze od nich np. $a_1 = 2$, $a_2 = 1$, $a_3 = 3$, odpowiednio. Powyższe twierdzenie mówi, że można znaleźć (jedyną) liczbę a mniejszą od $n = 3 \cdot 4 \cdot 5 = 60$ taką, że

$$a \equiv_3 2, \quad a \equiv_4 1, \quad a \equiv_5 3.$$

Liczymy m_i:

$$m_1 = \frac{60}{3} = 20, \quad m_2 = \frac{60}{4} = 15, \quad m_3 = \frac{60}{5} = 12.$$

i odwrotności m_i modulo n_i:

$$m_1 \equiv_3 20 \equiv_3 2$$

$$2 \cdot 2 \equiv_3 1$$

czyli odwrotnością 2 modulo 3 jest 2, tzn. $2^{-1} \bmod 3 = 2$.

Podobnie obliczmy, że

$$m_2 \equiv_4 15 \equiv_4 3, \quad 3^{-1} \bmod 4 = 3,$$

$$m_3 \equiv_5 12 \equiv_5 2, \quad 2^{-1} \bmod 5 = 3.$$

Następnie obliczamy $c_i = m_i(m_i^{-1} \bmod n_i)$:

$$c_1 = 20 \cdot 2 = 40, \quad c_2 = 15 \cdot 3 = 45, \quad c_3 = 12 \cdot 3 = 36.$$

A na koniec obliczamy szukaną liczbę a:

$$a = (a_1c_1 + \ldots + a_kc_k) \bmod n = 2 \cdot 40 + 1 \cdot 45 + 3 \cdot 36 \bmod 60 =$$

$$= 80 + 45 + 108 \bmod 60 = 233 \bmod 60 = 53.$$

I jak łatwo sprawdzić $a_i \equiv_{n_i} a$:

$$53 \equiv_3 2, \quad 53 \equiv_4 1, \quad 53 \equiv_5 3.$$

3 Poprawność protokołu RSA

Pokażemy teraz, że protokół RSA jest poprawny to znaczy, że zachodzi następujący fakt.

Fakt 3.1 *Niech $e, d, n \in N$ będą liczbami opisanymi w protokole RSA. Wtedy funkcje $P, S : Z_n \longrightarrow Z_n$ opisane w protokole RSA są do siebie wzajemnie odwrotne.*

Dowód: Pokażemy, że dla dowolnego $M \in Z_n$

$$P(S(M)) = S(P(M)) = M.$$

Z opisu protokołu RSA wynika, że

$$P(S(M)) = S(P(M)) = M^{ed} \bmod n.$$

Ponieważ e jest odwrotnością d modulo $\phi(n) = (p-1)(q-1)$ to istnieje $k \in Z$ takie, że

$$ed = 1 + k(p-1)(q-1)$$

Jeżeli $M \equiv_p 0$ to też $M^{ed} \equiv_p 0$. Zatem w tym przypadku $M^{ed} \equiv_p M$.

Jeżeli $M \not\equiv_p 0$ to używając Małego Twierdzenia Fermata otrzymujemy

$$M^{ed} \equiv_p M^{1+k(p-1)(q-1)} \equiv_p M(M^{(p-1)})^{k(q-1)} \equiv_p M(1)^{k(q-1)} \equiv_p M$$

Podobnie można pokazać, że $M^{ed} \equiv_q M$, dla dowolnego $M \in Z_n$. Zatem z Twierdzenia chińskiego o resztach otrzymujemy, że $M^{ed} \equiv_n M$, dla dowolnego $M \in Z_n$.

Q.E.D.

4 Probabilistyczne znajdowanie liczb pierwszych.

Teraz możemy opisać probabilistyczny test pierwszości liczb. Procedura `swiadek(a,n)` sprawdza czy `a` jest świadkiem (nie dzielnikiem!) złożoności liczby `n`. Procedura oblicza czy dla `a` i `n` zachodzi równość z Małego Twierdzenia Fermata (Twierdzenie 2.5)i przy okazji sprawdza czy 1 nie ma nietrywialnych pierwiastków modulo n, zob. Twierdzenie 2.6.

```
function swiadek(a,n):boolean;
 begin
```

```
  Obliczamy tablice B dlugosci k
  reprezentującą liczbę n-1 binarnie;
  d:=1; i:=k; OK:=false;
  while (i>0) and not OK do begin
    x:=d;
    d:=d*d mod n;
    if (d=1) and (x<>1) and (x<>n-1) then OK:=true;
      if B[i]=1 then d:=d*a mod n;
      i:=i-1;
   end;
   swiadek:=OK or (d<>1);
 end;
```

Poniższa procedura używa procedury swiadek s razy by sprawdzić czy n jest liczbą złożoną.

```
function pierwsza (n,s):boolean;
 begin pierwsza:=true;
  for i:=1 to s do begin
   a:=random(1,n-1); {a jest liczba losowa pomiedzy 1 i n-1}
   if swiadek(a,n) then pierwsza:=false;
  end;
 end;
```

Procedura `pierwsza` losowo wybiera s razy potencjalnych świadków złożoności liczby n. Jeśli znajdzie choć jednego to odpowiedź jest `false` i liczba n jest złożona, a jeśli nie znajdzie świadka po s próbach to odpowiedź jest `true` i mamy pewne szansę na to, że liczba jest jednak pierwsza. By wiedzieć jaka to jest szansa należy wiedzieć ile jest świadków złożoności. O tym mówi poniższe twierdzenie.

Twierdzenie 4.1 *Jeśli n jest nieparzystą liczbą złożoną to liczba świadków złożoności n jest co najmniej $\frac{n-1}{2}$.*

Zatem, jeśli odpowiedź jest `true` to szansa na to, że liczba n jest złożona jest równa $\frac{1}{2^s}$.

Pozostaje jeszcze drobny problem z tym czy łatwo jest 'trafić' na dużą liczbę pierwszą. Poniższe twierdzenie mówi, że tak.

Twierdzenie 4.2 *Niech $\pi(n)$ będzie liczbą liczb pierwszych niewiększych niż n. Wtedy*

$$\lim_{n\to\infty} \frac{\pi(n)}{n/\ln(n)} = 1$$

Twierdzenie to mówi, że funkcja $n/\ln(n)$ dobrze przybliża funkcję $\pi(n)$. Czyli, że losowa liczba n ma szanse $\frac{1}{\ln(n)}$ być liczbą pierwszą. Na przykład $\ln(10^{100}) \approx 230$. Zatem biorąc po uwagę tylko liczby nieparzyste jest duża szansa znaleźć liczbę pierwszą 100-cyfrową po mniej więcej 100 próbach.